**TUGAS PRAKTIKUM PBP**

**Jawablah pertanyaan di bawah ini! tuliskan hasil analisis terkait pertanyaan tersebut .**

1. Apa yang salah dengan potongan kode ini?

while () {

print ‘Enter a name: ‘;

chomp ($input = );

if ($input ne ‘’) {

$names++;

}

else { last; }

}

Jawaban :

Tidak ada <STDIN> untuk menginputkan inputannya.

2. Apa yang salah dengan potongan kode ini (Petunjuk: mungkin ada lebih dari satu kesalahan)?

if ($val == 4) then { print $val; }

elseif ($val > 4) { print “more than 4”; }

Jawaban :

Hilangkan “then” karena “then” bukan sintax perl dan e pada “elseif” karena seharusnya “elsif”.

3. Apa yang salah dengan potongan kode ini (Petunjuk: mungkin ada lebih dari satu kesalahan)?

for ($i = 0, $i < $max, $i++) {

$vals[$i] = “”;

}

Jawaban :

Koma di dalam perulangan seharusnya titik koma “;”.

4. Apa yang salah dengan potongan kode ini?

while ($i < $max) { $vals[$i] = 0; }

Jawaban :

Sintax error, Tidak mempunyai increment sehingga tidak memenuhi sebuah ketentuan dari kondisi loop.

5. Tulis ulang expressions berikut menggunakan splice:

push @list, 1;

push @list, (2,3,4);

$list[5] = “foo”;

shift @list;

Jawaban :

splice(@list, @list, 0, 1) atau splice(@list, $#list+1, 0, 1);

splice(@list, @list, 0, (2, 3, 4)) atau splice(@list, $#list+1, 0, (2,3,4));

splice(@list, 5, 1, "foo")

splice(@list, 0, 1)

6. Apa yang salah dengan potongan kode ini (Petunjuk: mungkin ada lebih dari satu kesalahan)?

while ($i <= $#list) {

$str = @novel[$i++];

push @final, reverse $str;

}

Jawaban :

Akses array yang digunakan disini ialah array splice, Ekspresi akses arraynya yang salah, karena akan berakhir di element bukan string sebenarnya. Lalu dalam line 3 dimana string tidak akan di reserve,kenyataannya reverse nya di panggil pada listnya. Jadi, string di artikan sebagai list of one element. Gunakan fungsi skalar untuk membuat scalar context untuk reverse.

7. Apa yang salah dengan skrip ini?

@int = &intersection(@list1, @list2);

sub intersection {

my (@1, @2) = @\_;

my @final = ();

OUTER: foreach my $el1 (@one) {

foreach my $el2 (@two) {

if ($el1 eq $el2) {

@final = (@final, $el1);

next OUTER;

}

}

}

return @final;

}

Jawaban :

Seharusnya @1 menjadi @one dan @2 menjadi @two. (pada penamaan variabel tidak boleh di awali oleh angka). Kesalahan lainnya ialah, mengasumsikan dua list @list1 dan @list2 adalah bagian dari subrutin. Padahal perl menggabungkan semua list argumen kedalam satu list.

8. Apa yang salah dengan skrip ini?

$numtimes = &search(@input, $key);

sub search { my (@in, $key) = @\_;

my $count = 0;

foreach my $str (@in) {

while ($str =~ /$key/og) { $count++; }

}

return $count;

}

Jawaban :

Karena order dari argumennya, list dan argumen skalarya di satukan menjadi satu kali ke subrutin. Kesalahannya dimana assign variabel @in dan $key ke argument list dalam @\_ terbalik, karena @in adalah list, maka ia akan mengcopy semua element dalam @\_ dan tidak akan menyisakan untuk $key. Urutkan argumen secara terbalik yaitu yang pertama key lalu list.